APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/082,417	02/26/2002	Vinit Jain	AUS920010895US1	4663

7590   11/06/2003

Mr. Volel Emile
P.O. Box 202170
Austin, TX 78720-2170

EXAMINER
PATEL, HETUL B

ART UNIT	PAPER NUMBER
2186	2

DATE MAILED: 11/06/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

	Application No.	Applicant(s)	
	10/082,417	JAIN ET AL.	
Office Action Summary	Examiner	Art Unit	
	Hetul Patel	2186	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>03</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>26 February 2002</u> .

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) <u>1-40</u> is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) <u>1-40</u> is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11)☐ The proposed drawing correction filed on _____ is: a)☐ approved b)☐ disapproved by the Examiner.

    If approved, corrected drawings are required in reply to this Office action.

12)☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____ .

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).

    a) ☐ The translation of the foreign language provisional application has been received.

15)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)	4) ☐ Interview Summary (PTO-413) Paper No(s). _____ .
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)	5) ☐ Notice of Informal Patent Application (PTO-152)
3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ .	6) ☐ Other: .

DETAILED ACTION

Specification

1.   The disclosure is objected to because of the following informalities:

Line 12 on page 12 should read "receiving partition uses the data" instead of "receiving

uses the data" as disclosed in this application.

Appropriate correction is required.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2.   Claims 1, 3, 7, 9, 13, 15, 19, 21, 25, 27, 29, 31, 33, 35, 37 and 39 are rejected

under 35 U.S.C. 103(a) as being unpatentable over Gulick et al. (USPN: 6,314,501),

hereinafter, Gulick.

With respect to claims 1 and 13, Gulick teaches a method, apparatus and

program code of transferring data from a first partition of partitioned computer system to

a second partition by passing a pointer to the buffer to the second partition (e.g. see

Col. 3, lines 6-25). Gulick does not disclose that before passing a pointer to the buffer to

the second partition marking that buffer containing the data, which is in the first partition,

as a "read-only" buffer. However, it would be obvious to one having ordinary skill in the

art at the time of the current invention was made to mark the buffer containing the data, which is in the first partition, as a "read-only" buffer in order to secure the data stored in that buffer from getting modified or deleted during the process of assigning it (the buffer) to the second partition, reading/copying the data from that buffer to the second partition and re-assigning that buffer back to the first partition.

With respect to the claims 3 and 15, Gulick discloses the claimed invention as described above. Gulick does not teach that before reading the data, the second partition ensures that the buffer containing the data is a "read-only" buffer. However, it would be obvious to one having ordinary skill in the art at the time of the current invention was made to ensure that before reading the data from the buffer, the buffer is a "read-only" buffer in order to secure the data stored in that buffer from getting modified or deleted during the process of reading/copying the data and re-assigning that buffer back to the first partition.

With respect to the claims 7 and 9, Gulick teaches a program code of transferring data from a first partition of a partitioned computer system to a second partition in the rejection of claims 1 and 3 above. It would have been obvious to one of ordinary skill in the art at the time of the current invention was made to generate a computer program product on a computer readable medium using this program/code so this program code can be either easily transferred to another multi-partitioned system or sold as a software package.

With respect to the claim 19, Gulick teaches a computer system being partitioned into a plurality of partitions (e.g. see the abstract) and being able to transfer data from a first partition to a second partition comprising:

-        at least one memory device for storing code data (e.g. see the abstract and 160 in Fig. 1); and

-        at least one processor for processing the code data (e.g. see the abstract and 110, 112, 114 in Fig. 1) to pass a pointer to the buffer to the second partition (e.g. see Col. 3, lines 6-25).

Gulick does not disclose that before the step of passing a pointer to the buffer to the second partition, there is a step of marking that buffer containing the data, which is in the first partition, as a "read-only" buffer. However, it would be obvious to one having ordinary skill in the art at the time of the current invention was made to mark the buffer containing the data, which is in the first partition, as a "read-only" buffer in order to secure the data stored in that buffer from getting modified or deleted during the process of assigning it (the buffer) to the second partition, reading/copying the data from that buffer to the second partition and re-assigning that buffer back to the first partition.

With respect to the claim 21, Gulick discloses the claimed invention as described above. Gulick does not teach that before reading the data, the second partition ensures that the buffer containing the data is a "read-only" buffer. However, it would be obvious to one having ordinary skill in the art at the time of the current invention was made to ensure that before reading the data from the buffer, the buffer is a "read-only" buffer in order to secure the data stored in that buffer from getting modified or deleted during the

process of reading/copying the data and re-assigning that buffer back to the first partition.

With respect to the claims 25 and 33, Gulick teaches a method, apparatus and program code for transferring data with the utmost security comprising:

-       storing the data in a buffer of a first partition of a partitioned computer system (e.g. see Col. 3, lines 17-23);

-       passing a pointer to the buffer to the second partition of the system (e.g. see Col. 3, lines 6-25).

Gulick does not disclose that before passing a pointer to the buffer to the second partition marking that buffer containing the data, which is in the first partition, as a "read-only" buffer. However, it would be obvious to one having ordinary skill in the art at the time of the current invention was made to mark the buffer containing the data, which is in the first partition, as a "read-only" buffer in order to secure the data stored in that buffer from getting modified or deleted during the process of assigning it (the buffer) to the second partition, reading/copying the data from that buffer to the second partition and re-assigning that buffer back to the first partition.

With respect to the claims 27 and 35, Gulick discloses the claimed invention as described above. Gulick does not teach that before reading the data, the second partition ensures that the buffer containing the data is a "read-only" buffer. However, it would be obvious to one having ordinary skill in the art at the time of the current invention was made to ensure that before reading the data from the buffer, the buffer is a "read-only" buffer in order to secure the data stored in that buffer from getting modified

or deleted during the process of reading/copying the data and re-assigning that buffer

back to the first partition.

With respect to the claims 29 and 31, Gulick teaches a program code of

transferring data from a first partition of a partitioned computer system to a second

partition in the rejection of claims 25 and 27 above. It would have been obvious to one

of ordinary skill in the art at the time of the current invention was made to generate a

computer program product on a computer readable medium using this program/code so

this program code can be either easily transferred to another multi-partitioned system or

sold as a software package.

With respect to the claim 37, Gulick teaches a computer system being partitioned

into a plurality of partitions (e.g. see the abstract) and being able to transfer data with

utmost security comprising:

-        at least one memory device for storing code data (e.g. see the abstract

         and 160 in Fig. 1); and

-        at least one processor for processing the code data (e.g. see the abstract

         and 110, 112, 114 in Fig. 1) to store the data in a buffer of a first partition

         of a partitioned computer system (e.g. see Col. 3, lines 17-23), and to

         pass a pointer to the buffer to the second partition of the system thereby

         transferring the data with the utmost security (e.g. see Col. 3, lines 6-25).

Gulick does not disclose that before the step of passing a pointer to the buffer to

the second partition, there is a step of marking that buffer containing the data, which is

in the first partition, as a "read-only" buffer. However, it would be obvious to one having

ordinary skill in the art at the time of the current invention was made to mark the buffer

containing the data, which is in the first partition, as a "read-only" buffer in order to

secure the data stored in that buffer from getting modified or deleted during the process

of assigning it (the buffer) to the second partition, reading/copying the data from that

buffer to the second partition and re-assigning that buffer back to the first partition.

With respect to the claim 39, Gulick discloses the claimed invention as described

above. Gulick does not teach that before reading the data, the second partition ensures

that the buffer containing the data is a "read-only" buffer. However, it would be obvious

to one having ordinary skill in the art at the time of the current invention was made to

ensure that before reading the data from the buffer, the buffer is a "read-only" buffer in

order to secure the data stored in that buffer from getting modified or deleted during the

process of reading/copying the data and re-assigning that buffer back to the first

partition.

3.   Claims 2, 4-6, 8, 10-12, 14, 16-18, 20, 22-24, 26, 28, 30, 32, 34, 36, 38 and 40

are rejected under 35 U.S.C. 103(a) as being unpatentable over Gulick in view of

Awada et al. (USPN: 2003/0131042), hereinafter, Awada.

With respect to claims 2 and 14, Gulick discloses the claimed invention as

described above. Gulick does not teach that upon passing the pointer to the buffer to

the second partition, the buffer is re-assigned to the second partition. However, Awada,

on the other hand, teaches that upon receiving the request, the control node verifies

whether the requesting partition has permission to use the buffer (the device). If so, the

buffer (the device) gets re-assigned to the requesting partition (e.g. see the abstract). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to implement the Gulick's method, apparatus and program code by adding a step so upon passing the pointer to the buffer to the second partition, the buffer is re-assigned to the second partition as taught by Awada to make that buffer "local" to the requesting partition. Once the buffer becomes "local" to the requesting partition, the requesting partition can read/copy the data from it a lot more quickly and securely comparing to reading/copying that data from the other partition. Awada also teaches that after the requesting partition has used the buffer (the device), the buffer (the device) gets re-assigned to the partition to which it was originally assigned (see the abstract, paragraph 12 and 36), based on this rationale, claims 4 and 16 are rejected.

With respect to claims 5 and 17, Gulick teaches a method, apparatus and program code of transferring data from a first partition of partitioned computer system to a second partition by passing a pointer to the buffer to the second partition (e.g. see Col. 3, lines 6-25). Gulick does not disclose that before passing a pointer to the buffer to the second partition marking that buffer containing the data, which is in the first partition, as a "read-only" buffer. However, it would have been obvious to one having ordinary skill in the art at the time of the current invention was made to mark the buffer containing the data, which is in the first partition, as a "read-only" buffer in order to secure the data stored in that buffer from getting modified or deleted during the process of assigning it (the buffer) to the second partition, reading/copying the data from that

partition to the second partition and re-assigning back to the first partition. Gulick also does not teach about re-assigning the buffer to the second partition after passing the pointer to the buffer to the second partition. However, Awada, on the other hand, teaches that upon receiving the request, the control node verifies whether the requesting partition has permission to use the buffer (the device). If so, the buffer (the device) gets re-assigned to the requesting partition (e.g. see the abstract). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to implement the Gulick's method, apparatus and program code by adding a step so upon passing the pointer to the buffer to the second partition, the buffer is re-assigned to the second partition as taught by Awada to make that buffer "local" to the requesting partition. Once the buffer becomes "local" to the requesting partition, the requesting partition can read/copy the data from it a lot more quickly and securely comparing to reading/copying that data from the other partition. Awada also teaches that after the requesting partition has used the buffer (the device), the buffer (the device) gets re-assigned to the partition to which it was originally assigned (see the abstract, paragraph 12 and 36), based on this rationale, claims 6 and 18 are rejected.

With respect to the claims 8 and 10, Gulick in view of Awada teaches a program code of transferring data from a first partition of a partitioned computer system to a second partition in the rejection of claims 2 and 4 above. It would have been obvious to one of ordinary skill in the art at the time of the current invention was made to generate a computer program product on a computer readable medium using this program/code

so this program code can be either easily transferred to another multi-partitioned system or sold as a software package.

With respect to the claim 11-12, Gulick in view of Awada teaches a program code of transferring data from a first partition of a partitioned computer system to a second partition in the rejection of claims 5-6 above. It would have been obvious to one of ordinary skill in the art at the time of the current invention was made to generate a computer program product on a computer readable medium using this program/code so this program code can be either easily transferred to another multi-partitioned system or sold as a software package.

With respect to claim 20, Gulick discloses the claimed invention as described above. Gulick does not teach that upon passing the pointer to the buffer to the second partition, the buffer is re-assigned to the second partition. However, Awada, on the other hand, teaches that upon receiving the request, the control node verifies whether the requesting partition has permission to use the buffer (the device). If so, the buffer (the device) gets re-assigned to the requesting partition (e.g. see the abstract). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to implement the Gulick's computer system by adding a step so upon passing the pointer to the buffer to the second partition, the buffer is re-assigned to the second partition as taught by Awada to make that buffer "local" to the requesting partition. Once the buffer becomes "local" to the requesting partition, the requesting partition can read/copy the data from it a lot more quickly and securely comparing to reading/copying that data from the other partition. Awada also teaches that after the

requesting partition has used the buffer (the device), the buffer (the device) gets re-assigned to the partition to which it was originally assigned (see the abstract, paragraph 12 and 36), based on this rationale, claim 22 get rejected.

With respect to claim 23, Gulick teaches a computer system being partitioned into a plurality of partitions (e.g. see the abstract) and being able to transfer data from a first partition to a second partition comprising:

-        at least one memory device for storing code data (e.g. see the abstract and 160 in Fig. 1); and

-        at least one processor for processing the code data (e.g. see the abstract and 110, 112, 114 in Fig. 1) to pass a pointer to the buffer to the second partition (e.g. see Col. 3, lines 6-25).

Gulick does not disclose that before the step of passing a pointer to the buffer to the second partition, there is a step of marking that buffer containing the data, which is in the first partition, as a "read-only" buffer. However, it would be obvious to one having ordinary skill in the art at the time of the current invention was made to mark the buffer containing the data, which is in the first partition, as a "read-only" buffer in order to secure the data stored in that buffer from getting modified or deleted during the process of assigning it (the buffer) to the second partition, reading/copying the data from that buffer to the second partition and re-assigning that buffer back to the first partition. Gulick also does not teach about re-assigning the buffer to the second partition after passing the pointer to the buffer to the second partition. However, Awada, on the other hand, teaches that upon receiving the request, the control node verifies whether the

requesting partition has permission to use the buffer (the device). If so, the buffer (the device) gets re-assigned to the requesting partition (e.g. see the abstract). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to implement the Gulick's method, apparatus and program code by adding a step so upon passing the pointer to the buffer to the second partition, the buffer is re-assigned to the second partition as taught by Awada to make that buffer "local" to the requesting partition. Once the buffer becomes "local" to the requesting partition, the requesting partition can read/copy the data from it a lot more quickly and securely comparing to reading/copying that data from the other partition. Awada also teaches that after the requesting partition has used the buffer (the device), the buffer (the device) gets re-assigned to the partition to which it was originally assigned (see the abstract, paragraph 12 and 36), based on this rationale, claim 24 get rejected.

With respect to claims 26 and 34, Gulick discloses the claimed invention as described above. Gulick does not teach that upon passing the pointer to the buffer to the second partition, the buffer is re-assigned to the second partition. However, Awada, on the other hand, teaches that upon receiving the request, the control node verifies whether the requesting partition has permission to use the buffer (the device). If so, the buffer (the device) gets re-assigned to the requesting partition (e.g. see the abstract). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to implement the Gulick's method, apparatus and program code by adding a step so upon passing the pointer to the buffer to the second partition, the buffer is re-assigned to the second partition as taught by Awada to make

that buffer "local" to the requesting partition. Once the buffer becomes "local" to the requesting partition, the requesting partition can read/copy the data from it a lot more quickly and securely comparing to reading/copying that data from the other partition. Awada also teaches that after the requesting partition has used the buffer (the device), the buffer (the device) gets re-assigned to the partition to which it was originally assigned (see the abstract, paragraph 12 and 36), based on this rationale, claims 28 and 36 are rejected.

With respect to the claims 30 and 32, Gulick in view of Awada teaches a program code of transferring data from a first partition of a partitioned computer system to a second partition in the rejection of claims 26 and 28 above. It would have been obvious to one of ordinary skill in the art at the time of the current invention was made to generate a computer program product on a computer readable medium using this program/code so this program code can be either easily transferred to another multi-partitioned system or sold as a software package.

With respect to claim 38, Gulick discloses the claimed invention as described above. Gulick does not teach that upon passing the pointer to the buffer to the second partition, the buffer is re-assigned to the second partition. However, Awada, on the other hand, teaches that upon receiving the request, the control node verifies whether the requesting partition has permission to use the buffer (the device). If so, the buffer (the device) gets re-assigned to the requesting partition (e.g. see the abstract). Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the current invention was made to implement the Gulick's computer system by adding a step so

upon passing the pointer to the buffer to the second partition, the buffer is re-assigned

to the second partition as taught by Awada to make that buffer "local" to the requesting

partition. Once the buffer becomes "local" to the requesting partition, the requesting

partition can read/copy the data from it a lot more quickly and securely comparing to

reading/copying that data from the other partition. Awada also teaches that after the

requesting partition has used the buffer (the device), the buffer (the device) gets re-

assigned to the partition to which it was originally assigned (see the abstract, paragraph

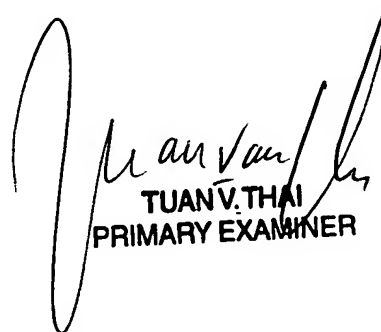12 and 36), based on this rationale, claim 40 get rejected.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Hetul Patel whose telephone number is (703) 305-6219. The examiner can normally be reached on M-F 8-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matt Kim can be reached on (703) 305-3821. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

HBP

TUAN V. THAI
PRIMARY EXAMINER